อาจหาญ สัตยารักษ์

SOFT MAIL

การพิมพ์อักขระแบบกราฟิก ตอนที่ 2

ชุดคำสั่งในการพิมพ์ภาษาไทยในเครื่องพิมพ์ 24 เข็ม

จากตอนที่ 1 ในไมโครฯ ฉบับที่ 63 ผู้อ่านก็ได้ความรู้ในการพิมพ์อักขระแบบกราฟิกบนเครื่องพิมพ์แบบ 9 เข็มไปแล้ว มาตอนที่ 2 นี้เราจะมาพูดถึงการพิมพ์แบบกราฟิกด้วยเครื่องพิมพ์ 24 เข็มกันบ้าง โดยเราจะมาศึกษาแนวคิดของการพิมพ์แบบกราฟิกบนเครื่อง 24 เข็มก่อนว่าเป็นอย่างไร

จากตารางแสดงการพิมพ์แบบกราฟิกในรูปที่ 1 (ตอนที่ 1) จะเห็นว่าเครื่องพิมพ์ 24 เข็มนั้นพิมพ์แบบกราฟิกได้ 5 แบบ คือ โหมด 32, 33, 38, 39 และ 40 ซึ่งจะมีความละเอียดต่างกันตามจำนวนจุดต่อหนึ่งนิ้วที่ปรากฏในตารางดังกล่าว โดยมีโหมด 40 พิมพ์ได้ขนาดเล็กที่สุดคือ 360 จุดต่อนิ้ว โหมด 39 พิมพ์ได้กว้างกว่าโหมด 40 สองเท่า

เลือกโหมดเพื่อใช้พิมพ์

เมื่อเรามีความประสงค์จะให้การพิมพ์สามารถพิมพ์ตัวปกติและตัวขยายได้อย่างสมบูรณ์ เราก็ต้องเลือกโหมดขนาดเล็กที่สุดมาใช้งานพิมพ์ตามปกติ นั่นคือเมื่อพิจารณาตามตารางที่ 1 เราก็จะใช้โหมด 40 และ 39 มาใช้ในการพิมพ์แบบกราฟิกตามที่เราต้องการ

มาดูจำนวนจุดต่อบรรทัด (กระดาษกว้าง 8 นิ้ว) จะพบว่าโหมด 40 พิมพ์ได้ 2,880 จุด ดังนั้นถ้าต้องการพิมพ์ตัวอักขระให้ได้ 80 ตัวอักขระในหนึ่งบรรทัด (10 ตัวต่อนิ้ว) ต้องใช้จำนวนจุด 2880/80 = 30 จุดต่อหนึ่งตัวอักขระ หรือ ถ้าต้องการตัวอักษรขนาด 96 ตัวอักขระต่อบรรทัด (12 ตัวต่อนิ้ว) ต้องใช้จำนวนจุดต่อหนึ่งอักขระ 2880/96 = 30 จุด

เพื่อสะดวกต่อการพิมพ์อักขระได้ทั้งแบบ 10 ตัวต่อนิ้วและแบบ 12 ตัวต่อนิ้ว โดยใช้อักขระชุดเดียวกัน ในการสร้างรูปแบบของอักขระแต่ละตัวจึงจะทิ้งช่องว่างไว้ที่หัวและท้าย เช่นด้านหัวว่างไว้ 4 จุดและด้านท้ายว่างไว้ 2 จุด รวมเป็นว่างไว้ 6 จุดซึ่งจะได้ส่วนที่เป็นอักขระแท้ ๆ 30 จุดพอดีที่จะใช้พิมพ์แบบ 12 ตัวต่อนิ้ว

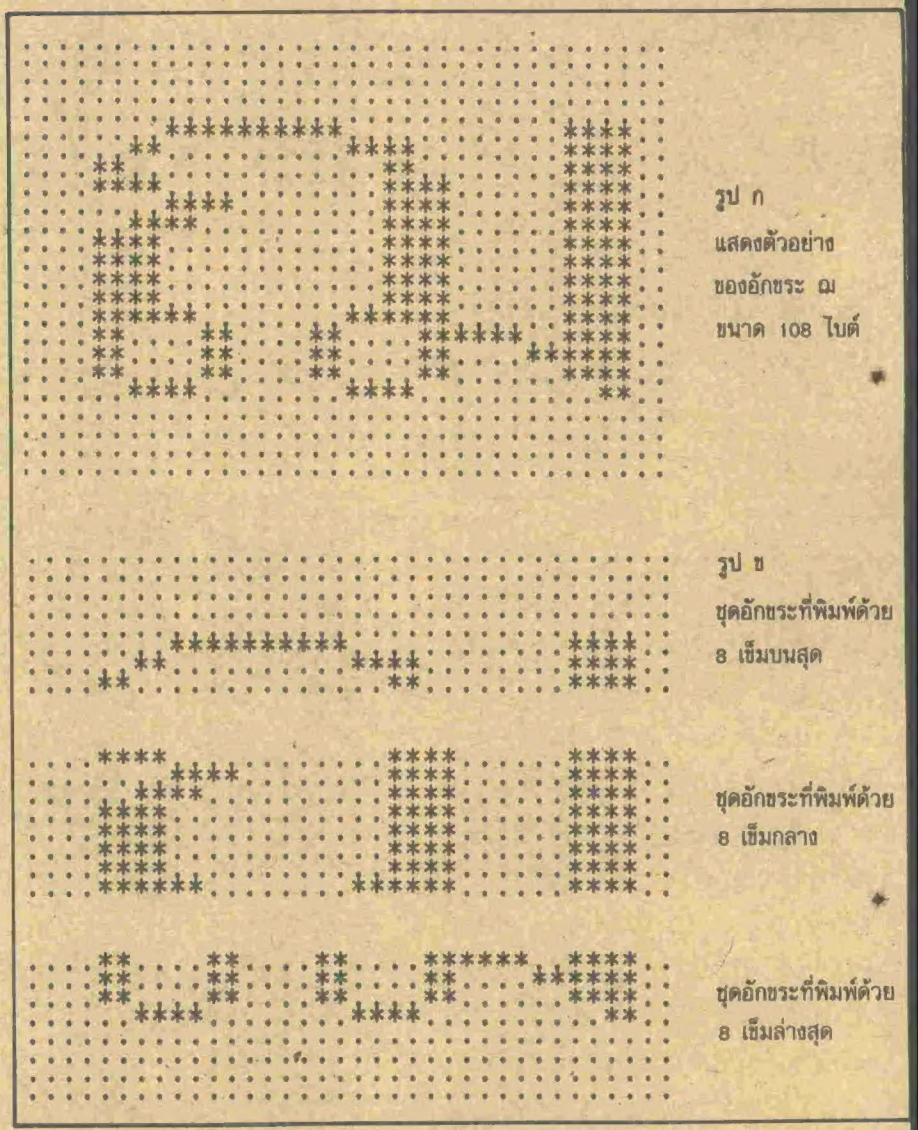
การพิมพ์แบบกราฟิกของเครื่องพิมพ์ 24 เข็ม

การพิมพ์แบบกราฟิกของเครื่องพิมพ์ 24 เข็มมีความแตกต่างจากเครื่องพิมพ์ 9 เข็ม คือ

1. เครื่อง 9 เข็มควบคุมการพิมพ์ในแนวแนวตั้ง (1 แนว) ด้วยรหัส 8 บิต จำนวน 1 ไบต์ ส่วนเครื่องพิมพ์ 24 เข็มควบคุมการพิมพ์ในแนวตั้ง (1 แนว) ด้วยรหัส 8 บิตจำนวน 3 ไบต์ โดยควบคุมเข็มทีละ 8 เข็มเรียงจากบนลงล่างจำนวน 3 ชุด (8×3 = 24) ซึ่งทำให้ข้อมูลของหนึ่งอักขระในเครื่องพิมพ์ 24 เข็มต้องมีจำนวนทั้งสิ้น 36×3 = 108 ไบต์

2. จากข้อ 1 ทำให้ข้อมูลที่ใช้พิมพ์ของอักขระในเครื่อง 9 เข็มนั้นเรียงลำดับไปจากต้นไปท้าย ส่วนของเครื่อง 24 เข็มนั้น ข้อมูลลำดับที่ 1, 4, 7,10,13,...ใช้ควบคุมเข็มพิมพ์ 8 เข็มบนสุด ข้อมูลที่ 2, 5, 8, 11, 14,... ใช้ควบคุมเข็มพิมพ์ 8 เข็มกลาง และข้อมูลที่ 3, 6, 9, 12, 15,...ใช้ควบคุมเข็มพิมพ์ 8 เข็มล่างสุด กล่าวคือของเครื่องพิมพ์ 24 เข็มนั้นจะแบ่งข้อมูลออกได้เป็น 3 ชุดเพื่อใช้ควบคุมการพิมพ์ของเข็มพิมพ์ 8 เข็มจำนวน 3 ชุดดังกล่าว (ขอให้ดูรูปที่ 11 ประกอบ)

3. การส่งรหัสเพื่อบอกให้เครื่องพิมพ์รับรู้จำนวนจุด (ต่อแนวตั้ง 1 แนว) มีจำนวนเท่ากันคือ คิดเฉพาะการพิมพ์ของเข็ม 8 เข็มหนึ่งชุดเท่ากัน นั่นคือ ถ้าเราพิมพ์ตัวขนาด 10 ตัว/นิ้ว ก็ต้องบอกให้รับรู้ว่าจะพิมพ์ 36 แนวตั้งต่อหนึ่งตัวอักขระ และ 30 แนวตั้งต่อหนึ่งตัวอักขระสำหรับตัวขนาด 12 ตัว/นิ้ว เป็นต้น



รูปที่ 11 รูปแสดงอักขระที่ใช้ในการพิมพ์แบบกราฟิกของเครื่องพิมพ์ 24 เข็ม

4. ระยะการเลื่อนกระดาษของเครื่องพิมพ์ 9 เข็มเป็น n/216 นิ้ว แต่สำหรับเครื่องพิมพ์ 24 เข็มกลับเป็น n/180 นิ้ว ดังนั้นการกำหนดระยะของการเลื่อนบรรทัดต้องเปลี่ยนไปให้เหมาะสมด้วย

เพื่อความสะดวกผู้เขียนจึงขอใช้รูปแบบของอักขระที่มีอยู่แล้วในราชวิถี WORD PC คือ FONT1.LQ1 และ FONT2.LQ2 มาใช้ในการพิมพ์ จึงต้องขอขอบคุณ นพ.ชุษณะ มะกรสาร มาพร้อมนี้ด้วย

โดยแต่ละไฟล์นั้นจะเก็บรหัส 224 ตัวอักขระ ๆ ละ 108 ไบต์โดยเรียงรหัสเหมือนกับของเครื่องพิมพ์ 9 เข็มที่เคยเขียนถึงในบทความเรื่อง การทำอักขระกราฟิกด้วย WORD PC ในฉบับที่ 52 ดังนั้นโปรแกรมของเราจึงเริ่มตามรูปที่ 12 ซึ่งเป็นส่วนหนึ่งในไฟล์ BIT24.PAS ที่อยู่ในซอฟต์เมล์

โดยการกำหนดตัวแปรเพื่อใช้ควบคุมการพิมพ์แบบปกติ, ตัวขยาย, ตัวหนา หรือการขีดเส้นใต้ ฯลฯ ไว้ด้วยในชุดคำสั่ง SaveFontLQ1; ได้กำหนด

ตัวแปรที่ใช้งานให้เป็นค่าเริ่มต้น และอ่านรูปแบบของอักขระทั้งสองชุดมาไว้ในตัวแปรมิติFont1 และ Font2

ส่วนคำสั่ง CopyByte นั้นก็เหมือนกับที่ใช้ในเครื่องพิมพ์ 9 เข็ม

การพิมพ์อักขระตามปกติ

ในการพิมพ์ตัวอักขระตามปกตินั้นจะเป็นไปดังชุดคำสั่ง CopyOneFont (ในรูปที่ 13) ซึ่งคราวนี้ขอเขียนในอีกรูปแบบหนึ่ง เพราะของเครื่อง 9 เข็มนั้นใช้วิธีคำนวณจำนวนจุดที่จะพิมพ์แบบกราฟิกรวมกันทีเดียวทั้งบรรทัดเลย แต่โปรแกรมของเครื่อง 24 เข็มนี้ผู้เขียนเปลี่ยนมาเป็นคำนวณทีละหนึ่งอักขระ (อาจจะพิมพ์ช้าอยู่บ้าง แต่ก็จะได้แนวคิดใหม่) รวมทั้งการพิมพ์ตัวขยายก็ใช้วิธีต่างจากของเดิมที่ใช้วิธีพิมพ์ด้วยโหมดการพิมพ์กราฟิกเดิมตลอด โดยพิมพ์ข้อมูลซ้ำสองครั้งมาเป็นการกำหนดโหมดจากพิมพ์จากโหมด 40 มาเป็นโหมด 39 ซึ่งเราไม่ต้องสั่งพิมพ์ข้อมูลซ้ำ

การพิมพ์ตัวหนาในเครื่องพิมพ์ 24 เข็มต้องเข้าใจการเรียงลำดับของข้อมูลประกอบ เพราะข้อมูลที่ 4 จะอยู่หลังข้อมูลที่ 1 (สำหรับ 8 เข็มบนสุด) ดังนั้นตอนที่จะพิมพ์ตัวหนาจึงต้องใช้ข้อมูลที่อยู่ข้างหน้าข้อมูลนั้นไปอีก 3 ข้อมูล (ในโปรแกรมคือ Font1[ASCII-32] [x-3] หรือ Font2[ASCII-32] [x-3]) มารวมกับข้อมูลที่คำนวณได้แบบ or เพื่อจะได้รูปแบบการพิมพ์แบบตัวหนา ซึ่งเข้า-ออกด้วย ^PB

ในกรณีที่มีการขีดเส้นใต้ (ค่าตัวแปร UnderLine=1) และเป็นการพิมพ์กลุ่มข้อความที่เป็นสระล่าง คือ อุ อู (ตรวจได้จาก RRound=2) จะเห็นได้ว่ายุ่งยากกว่าของเครื่องพิมพ์ 9 เข็มเล็กน้อย เพราะการพิมพ์เส้นใต้ควรเกิดขึ้นเมื่อสั่งให้หัวเข็ม 8 เข็มบนสุดทำงานเท่านั้น (ตรวจจาก x mod 3=1) โดยทำการรวมข้อมูลนั้น ๆ กับ 48 โดยการรวมแบบ or เช่นกันซึ่งจะมีผลทำให้หัวเข็มที่คุมด้วยบิตที่ 5

```
PROGRAM PrinterBitImage_24_pins;
uses crt,dos,Printer;
Type Font=array [1..108] of byte;
var  FontFile:File;
     Font1,Font2:array[0..223] of Font;
     High,Enlarge,Conden,Bold,UnderLine,Double,UpDn,CMode,
     RRound,FontMode,LeftMargin:byte;
     ch,PPModeOld:char;

PROCEDURE SaveFontLQ1;
 var   size:word;
 begin
  clrscr; textcolor(0); TextBackground(7);
  writeln('                                          ');
  writeln('      Print by Bit Image(24 Pins)         ');
  writeln('  By Arthan Sattayarak  Lampang 52000     ');
  writeln('                                          ');
  textcolor(3); TextBackground(0);  Gotoxy(1,7);
  writeln('Loading FONT1.LQ1');
    Assign(FontFile,'b:font1.lq1');
    Reset(FontFile,1);
    size:=Filesize(FontFile);
    blockRead(FontFile,Font1,size);
    Close(FontFile);
    writeln('Loading FONT2.LQ1');
    Assign(FontFile,'b:font2.lq1');
    Reset(FontFile,1);
    size:=Filesize(FontFile);
    blockRead(FontFile,Font2,size);
    Close(FontFile);
    Enlarge:=0; Conden:=0; Bold:=0; FontMode:=1; LeftMargin:=1;
    UnderLine:=0; Double:=0; UpDn:=0; Cmode:=1; High:=0; PPModeOld:=#40;
 end;

 PROCEDURE CopyByte(ByteSt :string);
 var  Regs : Registers;  B: byte;
 begin
   for b:=1 to length(ByteSt) do
     begin
       Regs.AH := 0;          Regs.AL := ord(ByteSt[b]);
       Regs.DX := Pred(1);    Intr($17, Regs);
     end;
 end;
```

รูปที่ 12 ชุดคำสั่ง SaveFontLQ1 เก็บรูปแบบอักขระเข้าในตัวแปร Font1, Font2

```
PROCEDURE CopyOneFont(ASCII,CMode:byte; Pmode:char);
const XStart:array[0..1] of byte=(1,13);
        XEnd:array[0..1] of byte=(108,102);
      CountSt:array[0..1] of Char=(#36,#30);
var  x:integer;   ByteDisp:byte;
begin
 copybyte(#27'*'+PMode+CountSt[CMode]+#0);
 if ASCII=32 then ASCII:=160;
   for x:=XStart[CMode] to XEnd[CMode] do
    begin
     if FontMode=1 then ByteDisp:=Font1[AscII-32][x]
                   Else ByteDisp:=Font2[AscII-32][x];
     if (x>XStart[CMode]+2) and (bold=1) then
          if FontMode=1 then ByteDisp:=ByteDisp or Font1[AscII-32][x-3]
                        Else  ByteDisp:=ByteDisp or Font2[AscII-32][x-3];
    if (UnderLine=1) and (RRound=2) and (x mod 3=1) and (ASCII=160) then
                                        ByteDisp:=ByteDisp or 48;
          copyByte(chr(ByteDisp));
     end;
end;
```

รูปที่ 13 ชุดคำสั่ง CopyOneFont สำหรับพิมพ์หัวอักขระแบบปกติ

และ 4 ทำงาน จึงได้เส้นใต้ตามต้องการ การขีดเส้นใต้นี้ได้ตรวจสอบการเว้นเมื่อเป็น อุ อู ญ ฎ ฏ ฐ ฤ ให้เรียบร้อยแล้ว (ดังนั้นสำหรับเครื่องพิมพ์ 9 เข็มซึ่งผู้เขียนทิ้งไว้ให้ผู้อ่านลองแก้ไขนั้นก็ใช้หลักการนี้ได้เช่นกัน

การพิมพ์อักขระสูง 2 เท่า

เราสามารถใช้ข้อมูลตามปกตินี้มาพิมพ์ให้เป็นตัวอักษรสูง 2 เท่าได้โดยไม่ต้องแก้ไขข้อมูลแต่อย่างใดทั้งสิ้น เพราะเราทราบแล้วว่า การพิมพ์หนึ่งครั้งเครื่องพิมพ์ต้องการข้อมูล 8 บิตจำนวน 3 ไบต์เพื่อกำหนดให้เข็มทั้ง 24 เข็มทำงาน ดังนั้นเมื่อเรามีข้อมูลขนาด 24 บิต (3 ไบต์) เมื่ออยากจะทำให้เป็นตัวสูง 2 เท่าก็จะทำได้ดังนี้

1. นำข้อมูล 24 บิตมาแยกเป็น 2 ส่วน คือ 12 บิตสูง และ 12 บิตต่ำ

2. นำข้อมูล 12 บิตสูงมาขยายให้เป็นข้อมูล 24 บิตส่งไปพิมพ์ครั้งที่ 1 จะได้ตัวอักขระครึ่งบน ซึ่งจะสูงเท่ากับตัวอักขระที่พิมพ์ตามปกติ แล้วเลื่อนกระดาษขึ้น

3. นำข้อมูล 12 บิตต่ำมาขยายให้เป็นข้อมูล 24 บิต ส่งไปพิมพ์เป็นครั้งที่สอง จะได้ตัวอักขระครึ่งล่าง เมื่อรวมกับที่พิมพ์ไปแล้วก็จะได้ตัวอักขระสูง 2 เท่าตามต้องการ

การทำงานจริง ๆ ก็คือ ชุดคำสั่ง CopyOneFontHigh (ในรูปที่ 14) โดยจะทำการตรวจสอบข้อมูล 24 บิต ครั้งละ 12 บิต คือ

ครั้งแรกใช้ไบต์ที่ 1 (8 บิต) และไบต์ที่ 2 (4 บิตสูง) รวมเป็น 12 บิตสูง ครั้งที่สองใช้ไบต์ที่ 2 (4 บิตต่ำ) และไบต์ที่ 3 (8 บิต) รวมเป็น 12 บิตต่ำ หลังจากนั้นก็ใช้ไบต์ถัด ๆ ไปครั้งละ 3 ไบต์มาคำนวณเพื่อใช้พิมพ์ตัวสูง 2 เท่า

```
PROCEDURE CopyOneFontHigh(ASCII,CMode:byte; Pmode:char; Timeck:byte);
const XStart:array[0..1] of byte=(1,13);
        XEnd:array[0..1] of byte=(108,102);
      Countst:array[0..1] of char=(#36,#30);
      bit_High:array[0..7] of byte=(3,12,48,192,3,12,48,192);
      bitck:array[0..7] of byte=(1,2,4,8,16,32,64,128);

var  x,xbyte:integer; xx,ByteHigh,byteLow,ByteDisp:byte;
     BBold:array[0..512] of byte; Bold_ck:word;

begin
    if ASCII=32 then ASCII:=160;
    copybyte(#27'*'+PMode+CountSt[CMode]+#0);
    x:=XStart[CMode]+timeck;   Bold_ck:=0;
   Repeat xbyte:=x;   ByteHigh:=0; ByteLow:=0;
       if FontMode=1 then ByteDisp:=Font1[AscII-32][xbyte]
                     Else ByteDisp:=Font2[AscII-32][xbyte];
     if ASCII <>160 then          begin
       for xx:=0 to 7 do
         begin
           if ByteDisp and Bitck[xx] =Bitck[xx] then
              if xx>3 then ByteHigh:=byteHigh+Bit_High[xx]
                           Else ByteLow:=ByteLow+Bit_High[xx]
         end;                    end;
        BBold[Bold_ck]:=ByteHigh;
        BBold[Bold_ck+1]:=ByteLow; Inc(Bold_ck,2);
      if (Bold=1) and (Bold_ck>5) then                 begin
           ByteHigh:=ByteHigh or BBold[Bold_ck-6];
           ByteLow:=ByteLow or BBold[Bold_ck-5];    end;
      if (UnderLine=1) and (RRound=2) and (ASCII=160) then
                       ByteHigh:=ByteHigh or 12;
      if Timeck=0 then copyByte(chr(ByteHigh)+chr(ByteLow))
                  Else  copyByte(chr(ByteLow));
        Inc(xbyte);
        ByteDisp:=Font1[AscII-32][xbyte]; ByteHigh:=0; ByteLow:=0;
     if ASCII <>160 then          begin
       for xx:=0 to 7 do
         begin
           if ByteDisp and Bitck[xx] =Bitck[xx] then
                if xx>3 then ByteHigh:=ByteHigh+Bit_High[xx]
                             Else ByteLow:=ByteLow+Bit_High[xx]
         end;                    end;
        BBold[Bold_ck]:=ByteHigh;
        BBold[Bold_ck+1]:=ByteLow; Inc(Bold_ck,2);
      if (Bold=1) and (Bold_ck>5) then                 begin
           ByteHigh:=ByteHigh or BBold[Bold_ck-6];
           ByteLow:=ByteLow or BBold[Bold_ck-5];    end;
      if Timeck=0 then  copyByte(chr(ByteHigh))
                  Else copyByte(chr(ByteHigh)+chr(ByteLow));
        Inc(x,3);
   Until x>XEnd[CMode];
end;
```

รูปที่ 14 ชุดคำสั่ง CopyOneFontHigh สำหรับพิมพ์ตัวอักขระสูง 2 เท่า

มีตัวแปรที่นำมาใช้คือ ByteHigh, ByteLow โดยตามโปรแกรมได้ทำการเปลี่ยนข้อมูล 8 บิตตามปกตินั้นให้มาเป็นข้อมูล 16 บิตคือ ByteHigh และ ByteLow โดยการพิมพ์อักขระแต่ละตัวจะพิมพ์ 2 ครั้งซึ่งครั้งแรก (อักขระครึ่งบน) จะส่งรหัสให้แก่หัวเข็ม 24 เข็มตามรหัส ByteHigh กับ ByteLow ของไบต์ที่ 1 และรหัส ByteHigh ของไบต์ที่ 2 ไปให้ ส่วนครั้งที่สอง (อักขระครึ่งล่าง) นั้นส่งรหัส ByteLow ของไบต์ที่ 2 และ ByteHigh กับ ByteLow ของไบต์ที่ 3 ไปพิมพ์ ในชุดคำสั่งของการพิมพ์ขนาดสูง 2 เท่านี้ยังสามารถพิมพ์แบบตัวหนา และขีดเส้นใต้ได้

การพิมพ์ในหนึ่งบรรทัด

การพิมพ์แบบกราฟิกของเครื่องพิมพ์ 24 เข็มนี้ผู้เขียนได้ทำการตรวจสอบการพิมพ์ต่างไปจากเครื่องพิมพ์ 9 เข็มทั้งการพิมพ์แบบขยาย หรือพิมพ์

ตัวอักขระที่ได้รูปแบบแปลกออกไปจากเดิม จึงขอให้ศึกษาโปรแกรมได้จากชุดคำสั่ง CopyOneLine (รูปที่ 15) ซึ่งใช้ตัวแปรดังนี้ คือ

ตัวแปร PPmode สำหรับส่งไปควบคุมรูปแบบของตัวอักษรที่แสดงออกทางเครื่องพิมพ์ ซึ่งจะเห็นว่าได้ใช้โหมดการพิมพ์แบบกราฟิกครบทั้ง 5 โหมดจึงทำให้ได้อักขระได้หลายรูปแบบ

ตัวแปร Cmode นั้นใช้ควบคุมขนาดตัวอักษรซึ่งในกรณีนี้จะใช้ รหัส #1(^PA) สำหรับตัวอักขระขนาด 12 ตัว/นิ้ว และ #14(^PN) สำหรับ 10 ตัว/นิ้ว

การตรวจข้อความถ้าพบรหัสแอสกีน้อยกว่า 32 ก็จะใช้เป็นตัวควบคุณลักษณะของตัวอักษร เช่น

#2(^PB) ใช้เข้า-ออกตัวหนา

#5(^PE) ใช้เข้า-ออกตัวขยาย

#23(^PW) ใช้เข้า-ออกตัวกว้างแบบที่ 1

#12(^PL) ใช้เข้า-ออกตัวกว้างแบบที่ 2

#17(^PQ) ใช้เข้า-ออกตัวกว้างแบบที่ 3

#19(^PS) ใช้เข้า-ออกการขีดเส้นใต้

#28(^P\, ^P1) เลือกใช้ font ที่ 1

```
PROCEDURE CopyOneLine(content:string);
var x:word;  PPmode:char;  Timeck:Byte;

begin
   copybyte(#27#108+chr(leftMargin)+#27#51#19);
   PPMode:=PPModeOld; Timeck:=0;
Repeat  x:=1;
 Repeat
  Case content[x] of
   #1:Cmode:=1;
  #14:Cmode:=0;
   #2:begin Inc(Bold); Bold:=Bold mod 2;  end;
   #5:begin Inc(Enlarge); Enlarge:=Enlarge mod 2;
        case Enlarge of
          1:begin PPmodeOld:=PPMode;
                case PPMode of
                  #40:PPMode:=#39;
                  #39:PPMode:=#33;
                  #33:PPMode:=#38;
                  #38:PPMode:=#32;
                end;
            end;
          0:PPMode:=PPModeOld;
        end;
      end;
   #19:begin Inc(UnderLine); UnderLine:=Underline mod 2; end;
    #4:PPmode:=#40;
   #12:PPMode:=#33;
   #17:PPMode:=#38;
   #23:PPMode:=#39;
   #28:FontMode:=1;
   #29:FontMode:=2;
   #32..#255:begin
              if High=0 then  copyOneFont(ord(Content[x]),CMode,PPmode)
               Else copyOneFontHigh(ord(Content[x]),CMode,PPmode,Timeck);
             end;
  end; { end case }
     Inc(x);
 Until x>length(Content);
  inc(Timeck);
   case High of
     0:begin CopyByte(#10); Timeck:=3; end;
     1:begin
        case Timeck of
          1:CopyByte(#27#51#22#10);
          2:begin  if RRound=2 then copyByte(#27#51#0#10)
                               Else  copyByte(#27#51#14#10);
            end;
        end;
       end;
   end;
Until Timeck>1;
end;
```

รูปที่ 15 ชุดคำสั่ง CopyOneLine สำหรับพิมพ์ข้อความหนึ่งบรรทัด

#28(^P\, ^P1) เลือกใช้ font ที่ 1

#29(^P], ^P2) เลือกใช้ font ที่ 2

รายละเอียดอื่น ๆ ของชุดคำสั่งนี้ กรุณาศึกษาด้วยตนเองให้เข้าใจด้วย

การจัดภาษาไทยให้เครื่องพิมพ์

โดยแนวคิดเดียวกันกับการพิมพ์แบบกราฟิกของเครื่องพิมพ์ 9 เข็ม ผู้เขียนใช้ชื่อชุดคำสั่งเดียวกันคือ CopyFont เช่นเดียวกัน

รายละเอียดดูในรูปที่ 16 ซึ่งคิดว่าคงทำความเข้าใจได้ไม่ยากเช่นกัน

#149– #159 คือ อักขระแสดงส่วนของตาราง

#171– #174 คือ อักขระ ญ ฎ ฏ ฐ

#207 คือ สระอำ

#210– #212 คือสระ โ ไ ใ

#215– #216 คือ สระ อุ อู

#217– #255 คือ สระบน และวรรณยุกต์

ผู้เขียนได้เพิ่มส่วนที่ต่างจากเครื่อง 9 เข็มคือ การตรวจเลขยกกำลัง (#20 หรือ ^PT) และเลขห้อยท้าย (#22 หรือ ^PV) ไว้ด้วย

การจัดสระและวรรณยุกต์ของภาษาไทยในหน่วยความจำให้ตรงกับการเก็บรูปแบบอักขระในการพิมพ์ก็ยังคงคล้ายกับของเดิมของเครื่องพิมพ์ 9 เข็ม คือ ชุดคำสั่ง Print (ดูรายละเอียดได้จากรูปที่ 7 ของตอนที่ 1 เพราะในตอนนี้ผู้เขียนไม่ได้นำมาแสดงให้ดู) และขอให้อ่านคำอธิบายจากฉบับที่ 52 เพิ่มเติมเช่นเดิม

ถึงตรงนี้เราก็ใช้ชุดคำสั่งที่ใช้งานสมบูรณ์พร้อมที่จะทำงานได้แล้ว โดยเก็บชุดคำสั่งเรียงลำดับตามที่ได้กล่าวมาแต่ต้น (นำชุดคำสั่ง Print จากตอนที่ 1 มาด้วย) จากนี้เราจะลองเขียนข้อความส่งไปให้พิมพ์ดูกันสัก 2 ชุด

การส่งข้อความไปพิมพ์

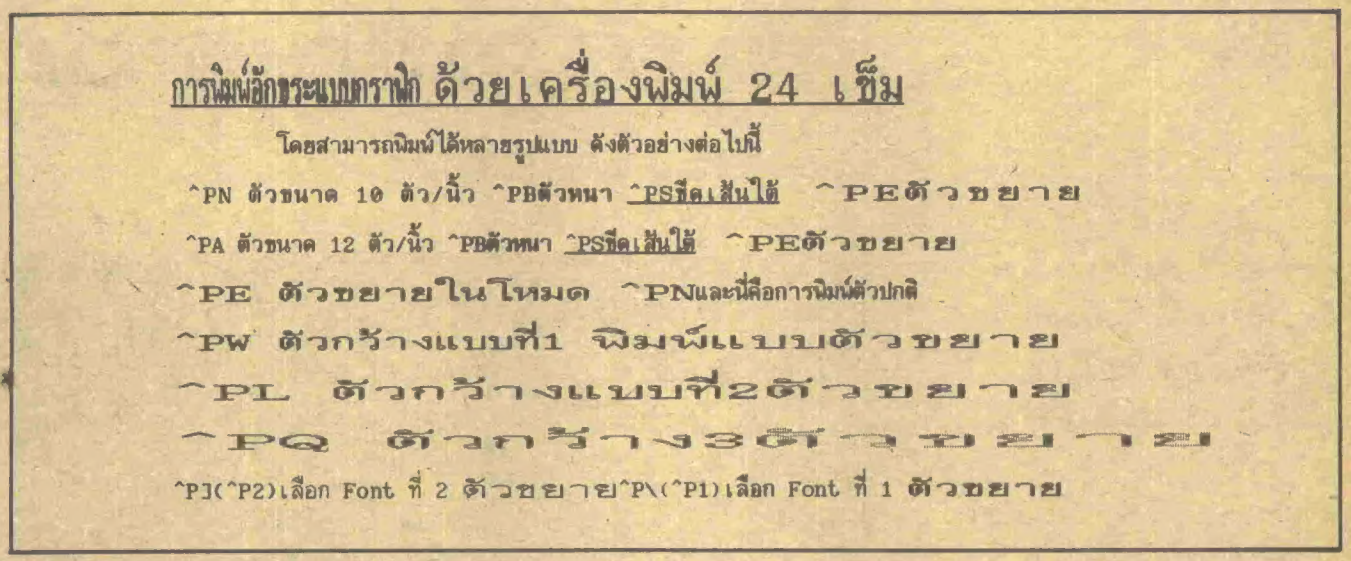
เราสามารถให้พิมพ์ข้อความได้ทั้งการกำหนดในปาสคาลเอง หรืออ่านมาจากที่พิมพ์ด้วยเวิร์ดตัวอื่นก็ได้ ขอให้ดูรูปที่ 17 ซึ่งมี 2 ชุดคำสั่ง คือ ชุด

```
PROCEDURE CopyFont(Content:string);
var Pword:array[0..2] of string;  R:word;

    Procedure SetCode(Code:string);
      var x:byte;
      begin
        for x:=0 to 2 do Pword[x]:=PWord[x]+Code;
      end;
begin
 For R:=0 to 2 do Pword[R]:='';
   if cmode=1 then SetCode(#1) Else SetCode(#14);
   R:=1;
 Repeat
   Case Content[R] of
     #1,#2,#4,#5,#12,#14,#17,#19,#23,#28,#29:SetCode(Content[R]);
     #20:begin Inc(UpDn); if UpDn=2 then UpDn:=0;  end;
     #8:High:=1;
     #22:begin Inc(UpDn);
               if UpDn=3 then UpDn:=0;  if UpDn=1 then UpDn:=2;
         end;
     #32..#148,#160..#170,#175..#206,#208,#209,#213,#214:begin
               if upDn=1 then Pword[0]:=Pword[0]+Content[R]
                         Else  Pword[0]:=Pword[0]+' ';
               if UpDn=0 then Pword[1]:=Pword[1]+Content[R]
                         Else  Pword[1]:=Pword[1]+' ';
               If UpDn=2 then  Pword[2]:=Pword[2]+Content[R]
                         Else  Pword[2]:=Pword[2]+' ';
                       end;
     #149..#159:begin if content[R] in [#152..#157,#159] then
                                        Pword[0]:=Pword[0]+#159;
                      Pword[1]:=Pword[1]+content[R];
                    if content[R] in [#149..#154,#159] then
                                        Pword[2]:=Pword[2]+#159;
                end;
     #171..#174:begin  Pword[0]:=Pword[0]+' ';
                       Pword[1]:=Pword[1]+Content[R];
                       Pword[2]:=Pword[2]+Chr(ord(Content[R])-36);
                end;
     #207:begin if Content[R-1]<#215 then
                  Pword[0]:=Copy(Pword[0],1,Length(Pword[0])-1)+#222+'
                  Pword[1]:=Pword[1]+#206; Pword[2]:=Pword[2]+' ';
          end;
     #210..#212:begin Pword[0]:=Pword[0]+chr(ord(Content[R])-78);
                      Pword[1]:=Pword[1]+#131;
                      Pword[2]:=Pword[2]+' ';
                end;
     #215,#216:Pword[2]:=Copy(Pword[2],1,Length(Pword[2])-1)+Content[R];
     #217..#255:Pword[0]:=Copy(Pword[0],1,Length(Pword[0])-1)+Content[R]
   end;
     Inc(R);
 Until R>Length(Content);
   for RRound:=0 to 2 do  CopyOneLine(PWord[RRound]);  High:=0;
END;
```

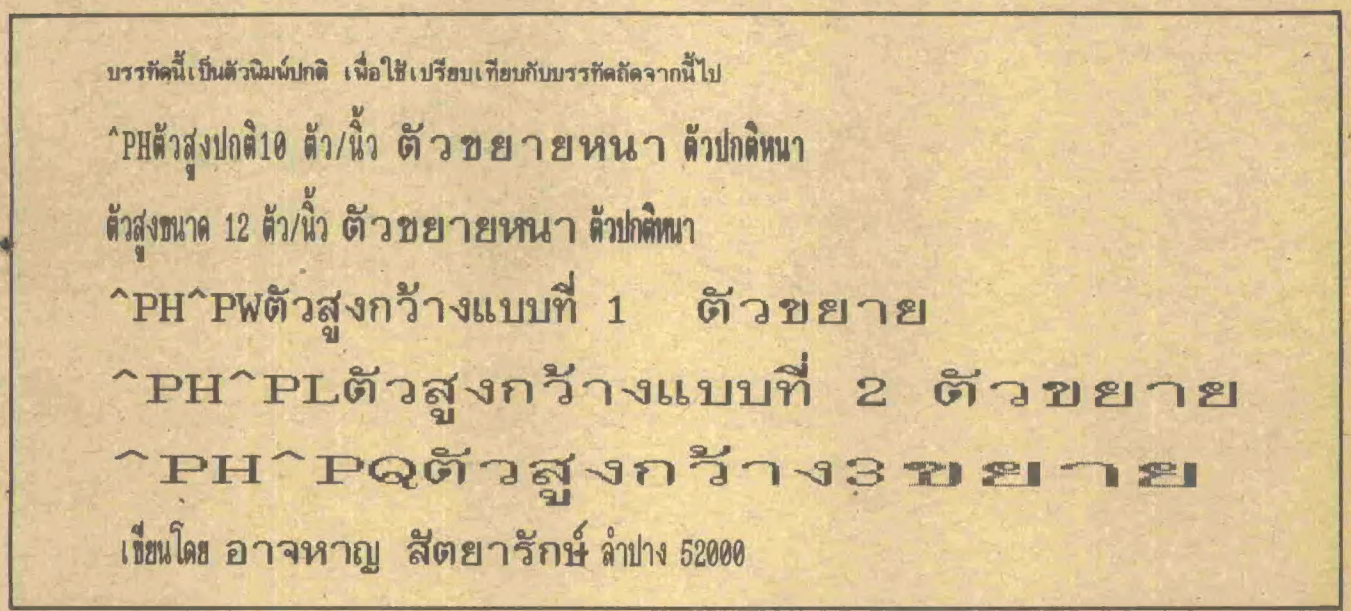
รูปที่ 16 ชุดคำสั่ง CopyFont สำหรับจัดข้อความให้การพิมพ์แบบกราฟิก

```
PROCEDURE Print1;
begin
  Print(#8#19'การพิมพ์'#2'อักขระแบบกราฟิก '#2#5+'ด้วยเครื่องพิมพ์ 24 เข็ม'#19#5);
  Print('          โดยสามารถพิมพ์ได้หลายรูปแบบ ดังตัวอย่างต่อไปนี้ ');
  Print(#14' ^PN ตัวขนาด 10 ตัว/นิ้ว '#2'^PBตัวหนา '#2#19+
                                      '^PBชิดเส้นได้'#19#5' ^PBตัวขยาย'#5);
  Print(#1' ^PA ตัวขนาด 12 ตัว/นิ้ว '#2'^PBตัวหนา '#2#19+
                                      '^PBชิดเส้นได้'#19#5' ^PBตัวขยาย'#5);
  Print(#5'^PE ตัวขยายในไหมด ^PN'#5'และนี่คือการพิมพ์ตัวปกติ');
  Print(#23'^PW ตัวกว้างแบบที่1 '#5'พิมพ์แบบตัวขยาย'#5);
  Print(#12'^PL ตัวกว้างแบบที่2'#5'ตัวขยาย'#5);
  Print(#17'^PQ ตัวกว้าง3'#5'ตัวขยาย'#5);
  Print(#4#29'^P](^P2)เลือก Font ที่ 2 '#5'ตัวขยาย'#5#28+
                                      '^P\(^P1)เลือก Font ที่ 1 '#5'ตัวขยาย'#5);
end;

PROCEDURE Print2;
 begin
  Print('บรรทัดนี้เป็นตัวพิมพ์ปกติ เพื่อใช้เปรียบเทียบกับบรรทัดถัดจากนี้ไป');
  Print(#8#4#14'^PHตัวสูงปกติ10 ตัว/นิ้ว '#5'ตัวขยาย'#2'หนา'#5' ตัวปกติหนา'#2);
  Print(#8#1'ตัวสูงขนาด 12 ตัว/นิ้ว '#5'ตัวขยาย'#2'หนา'#5' ตัวปกติหนา'#2);
  Print(#8#23'^PH^PWตัวสูงกว้างแบบที่ 1 '#5' ตัวขยาย'#5);
  Print(#8#12'^PH^PLตัวสูงกว้างแบบที่ 2 '#5'ตัวขยาย'#5);
  Print(#8#17'^PH^PQตัวสูงกว้าง3'#5#2'ขยาย'#2#5);
  Print(#8#4' เรียนโดย '#5'อาจหาญ สัตยารักษ์'#5' ลำปาง 52000');
 end;

{============================ MAIN ======================}
begin
  SaveFontLQ1;   clrscr;
  writeln('  ');
  writeln('กำลังพิมพ์ลงกระดาษ');
  Print1; Print2;
end.
```

รูปที่ 17 ชุดคำสั่ง Print1, Print2 ข้อความที่ใช้ทดลองพิมพ์



รูปที่ 18 ผลจากการพิมพ์แบบกราฟิกของชุดคำสั่ง Print1



รูปที่ 19 ผลจากการพิมพ์แบบกราฟิกของชุดคำสั่ง Print2

คำสั่ง Print1 และ Print2 อันเป็นชุดข้อความเพื่อส่งไปทดลองพิมพ์ รูปที่ 18 เป็นผลจากการพิมพ์ข้อความในชุดคำสั่ง Print1 และ รูปที่ 19 เป็นผลจากการพิมพ์ข้อความในชุดคำสั่ง Print2

ถึงตรงนี้ผู้อ่านก็ได้โปรแกรมสำหรับพิมพ์อักขระแบบกราฟิกของเครื่องพิมพ์ 24 เข็มไว้ใช้งานได้ตามที่ท่านต้องการเรียบร้อยแล้ว ท่านใดที่ต้องการรูปแบบอักขระที่แตกต่างไปจากนี้ก็ขอให้ลองใช้ความรู้พัฒนาดูก็จะสร้างความภูมิใจให้แก่ท่านมากยิ่ง ๆ ขึ้น

ผู้อ่านท่านใดที่ต้องการโปรแกรมพิมพ์อักขระที่อาศัย ROM ภาษาไทยรหัสเกษตร ซึ่งผู้เขียนพัฒนาในชื่อ WORD ลำปาง โปรดติดต่อได้ที่ผู้เขียน

พบกันใหม่คราวหน้า...